Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Research on dynamic grouping of heterogeneous agents for exploration and strike missions*

Chen CHEN[†1], Xiao-chen WU[‡1], Jie CHEN[1,2], Panos M. PARDALOS[‡3], Shu-xin DING[4]

[1]*School of Automation, Beijing Institute of Technology, Beijing 100081, China*
[2]*Department of Control Science and Engineering, Tongji University, Shanghai 200092, China*
[3]*Center for Applied Optimization, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA*
[4]*Signal and Communication Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing 100081, China*
[‡]E-mail: 3120160424@bit.edu.cn; pardalos@ufl.edu

Received mmm. dd, 2016; Revision accepted mmm. dd, 2016; Crosschecked mmm. dd, 2017

Abstract: Nowadays, the ever-changing environment and complex combat missions bring new demands on the formation of mission groups for unmanned combat agents. This paper aims to address the problem of the dynamic construction of mission groups under new requirements. Agents are heterogeneous, a group formation method is designed to dynamically form new groups under the circumstance that missions are constantly being explored. In our method, a group formation strategy combining heuristic rules and response threshold models is proposed to dynamically adjust the members of the mission group and adapt to the needs of new missions. In addition, the degree of matching between the mission requirements and the group capabilities, and the communication cost when the group is formed are used as indicators to evaluate the quality of the group. The response threshold method and the ant colony algorithm were selected as the comparison algorithm in the experiment. The results show that the grouping scheme obtained by the proposed method is superior to the comparison method.

Key words: Multi–agent; Dynamic missions; Group formation; Heuristic rule; Networking overhead
https://doi.org/10.1631/FITEE.1000000 **CLC number:** TP

1 Introduction

Nowadays, Multi-Agent Systems (MAS) are widely used to perform complex missions in different fields (Merabet et al., 2014), such as fire control and rescue missions or military detection and strike missions. The first problem to be solved in those missions is how to organize multiple agents to complete missions, that is, how to assign the overall missions to each agent and ensure that the agents effectively cooperate. Group formation has a great influence on the ultimate performance of the whole multi-agent system.

When agents have different abilities or play different roles, it is particularly important to form their groups according to the needs of the mission. The problem of finding a partition of agents set in groups such that some utility functions is maximized is known to be NP-hard concerning different utility functions (Gerkey and Mataric, 2004; Vig and Adams, 2006). In the field of artificial intelligence and cooperative systems, especially in distributed collaboration, experts and scholars have done a lot of research on the organizational structure and synergy of multi-agent systems. They have focused on

[‡] Corresponding author
* Project supported by the National Natural Science Foundation of China (No. 61773066) and the Foundation of China Academy of Railway Sciences Corporation Limited under Grant 2019YJ071.

topics such as the emergent rule theory (Murphey et al., 2002), the game theory (Pardalos et al., 2008), cooperative autonomous systems (Hirsch et al., 2008; Pardalos et al., 2013; Khoshnoud et al., 2019) and the hierarchical cooperation model (Hirsch et al., 2009; Butenko et al., 2013).

Many scholars have studied the self-organization or dynamic grouping of agents, among which the research results in the context of confrontation are rich, (Ducatelle et al., 2010; Skorobogatov et al., 2020; Singh et al., 2010; Necsulescu and Schilling, 2015; Liu et al., 2013) and (Orfanus et al., 2016) are all based on this background.

In terms of solutions, in addition to the classic models and the methods described above, some heuristic rules have also been used for the formation of agents groups and their mission allocation. (Ramchurn et al., 2010) and (Padmanabhan and Suresh, 2015) focus on solving the mission group formation problem by heuristic method, and (A.H. and Kashan, 2019), (Guo et al., 2020), and (Oh et al., 2018) design heuristic methods to deal with the mission allocation problem.

However, as the size of the agent community expands, the versatility of some heuristic methods becomes limited and no longer applies to more complex mission environments. In this case, people turn to the individual behavior of the natural community and its emerging group behavior and apply it to the agent system so that the individuals can spontaneously form groups to perform complex missions according to dynamic mission information. In this process, agents demonstrate greater self-organization, collaboration, and adaptability to the environment. For example, In (Yang et al., 2014) and (Khan et al., 2019), a special ant colony algorithm is used to solve the problem of the construction of intelligent dynamic alliance. In addition to the ant colony optimization (ACO), other bio-population-based heuristics have also been used in group formation problems. In (Manathara et al., 2011) and (George et al., 2010), the particle swarm algorithm and some new heuristic strategies are used to solve the problem of group formation.

As the scale of the task group formation problem continues to change, researchers have tried different methods to solve it. These existing methods have relatively good results when dealing with the dynamic grouping of a single type of agent. However, with the enrichment of the types of agents, the existing methods can no longer complete the grouping work based on the mission's requirements for heterogeneous agents. Therefore, in this paper, different from the existing results, the matching of heterogeneous agents capabilities with mission requirements and new evaluation criteria are emphatically considered in the grouping process.

This paper aims to solve the mission group formation problem of heterogeneous agents in the battlefield environment. Each mission has different priority and capability requirements, and requires different agents to implete cooperately. The ability requirement represents the minimum ability required to destroy the target. The purpose of the mission is to find the targets and eliminate them as soon as possible.

Fig. 1 briefly describes the process of collaborative mission execution by heterogeneous agents. In the left module, there are many different types of agents that need to form mission groups to perform mission one and two. When new missions are discovered(three and four), the agents must adjust the grouping pattern according to the new mission status, form a new topology structure, and adapt to the new mission requirements (as shown in the right figure). What we need to do is to design an algorithm to achieve dynamic grouping. In the algorithm, heuristic rules and the response threshold method are combined to form a hybrid grouping strategy, which ensures the realization of the above process.

Furthermore, when agents perform missions in the form of a group, this group is bound to exist in the form of a mobile network, which can be used to maintain information exchange among members of the group. However, excessive traffic will increase the network burden and the probability of the agent being detected by the enemy. Therefore, to avoid the undesirable consequences caused by too much information transmission, it should be considered to reduce unnecessary communication when designing the heuristic rules for grouping.

The main contributions of this paper can be summarized as follows. Firstly, a model has been established to describe the attributes of the mission and the agent, and the evaluation method of the grouping scheme was also given. Secondly, a hybrid algorithm combining heuristic rules and improved response threshold method has been designed to solve

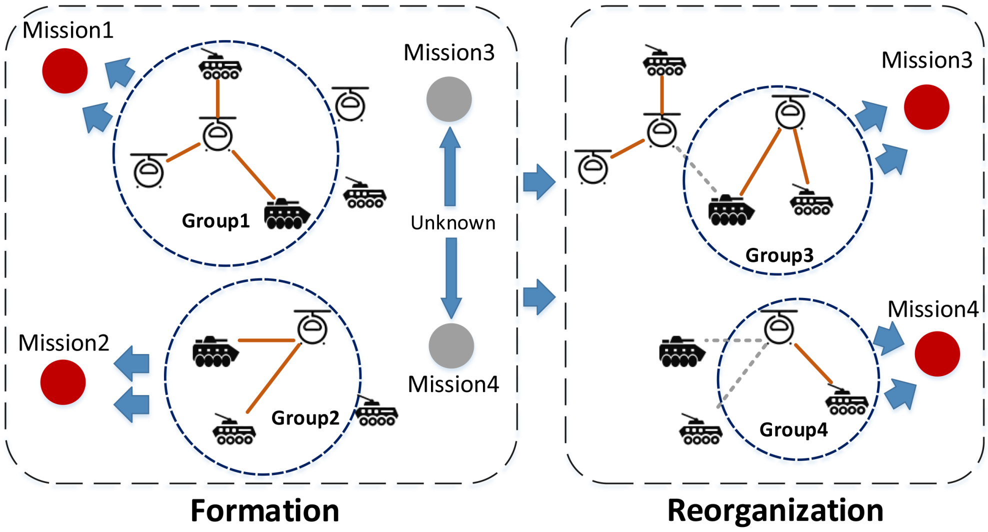


Fig. 1 Formation and reconstruction of mission groups

the agent dynamic grouping problem proposed in this paper.

The rest of this paper is organized below. The model is introduced in Section 2. Section 3 describes the dynamic grouping approach. The experiments are presented in Section 4. Lastly, conclusion and future work are given in Section 5.

2 Problem formulation

Our research focuses on the dynamic group formation problem, which is heterogeneous agents missioned with attacking some enemy targets. In the actual grouping process, a unified model is needed to accurately describe the status of the mission and the behavior and capabilities of the agents.

The model contains a mission area G, which is a rectangular, two-dimensional plane. In the area G, there are q enemy targets (Including stationary and moving targets) to be eliminated, and their initial positions are completely random. Each target is treated as a separate mission. Also, We have deployed p freely movable agents in the area to detect and strike targets. Since each mission requires agents with various capabilities to collaborate, different types of agents are Âăessential for grouping to accomplish the mission. When the mission situation changed based on the original grouping, the agents would reformation according to the new mission list. The agents need to eliminate enemy targets as much as possible.

2.1 Agents

- Definitions

First, we give the following definitions of the agents used in this paper:

(1) Each agent is a carrier of resource capabilities and a mission platform with certain autonomous capabilities.

(2) There are different types of agents with different capabilities.

(3) The number of agents is limited and it is not possible to perform all missions at the same time.

(4) Networking overhead will be generated when agents form new mission groups.

- Types and Topology

This study adopts two types of agents: the detection agent and the attack agent. The detection agents mainly conduct large-scale reconnaissance operations, discover new targets, and real-time updates of mission intelligence. The updated data served as the basis for the current attack agent grouping. As the name implies, the attack agents mainly implement the attack on the enemy target. The dynamic grouping method of these agents based on mission intelligence is our main research content. The attack agents can be divided into several types, as they own different capabilities to tackle with various types of

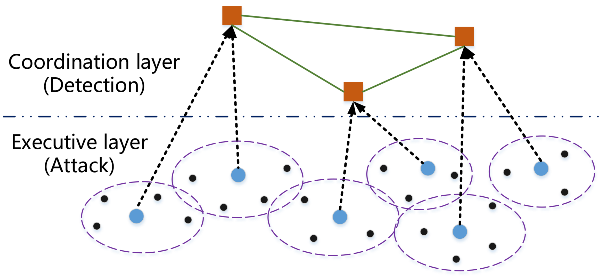


Fig. 2 Hierarchical topology

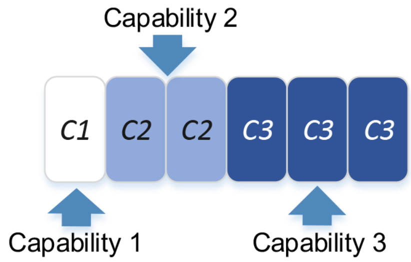


Fig. 3 Slot model

targets. Thus, we have to design a rational grouping method based on different types of agents to improve the efficiency of mission execution.

In the process of forming a mission group, a mobile network topology is built among agents, which is used for information exchange between individuals. The concept of hierarchical network is introduced into the system, and we achieve the three-dimensional topology through hierarchical modeling.

All the agents involved in the mission formed a hierarchical topology structure as shown in Fig. 2, which consists of two layers. The lower layer is the mission execution layer, which contains mission groups composed of attacking agents; the upper layer is the coordination layer, which contains only detection agents. In actual combat, a communication link is formed between the detection agent and the leader of the mission group. In addition to exploring the mission location and posting mission information, the detection agents also need to coordinate among the mission groups when the mission is released. If each mission group is regarded as a small network, the detection agent can be understood as a mobile gateway node, which is used for communication and coordination between networks.

- Capabilities

The agents' capabilities are described by a simple slot model, which has been used by researchers in the area of resource collection (Moritz and Middendorf, 2015). In these models, a slot is the smallest relevant unit of the agents' capabilities.

As Fig. 3 depicts, the slots of different colors represent the different capabilities of the agent. The number of types of slots represents the number of capabilities the agent owns. The different colors of slots represents the level of its ability. There are three capability slots in Fig. 3 to indicate that the agent has three different capabilities, and the capability values for each capability are shown on the right. p_{ij} represents the jth capability in agent i. When $p_{ij} > 0$, it means that agent i has the jth capability. In the problem we studied, the number of capability slots of the agent is different, and the value of each capability is also different.

- Constraints

Ignoring the impact of the environment, we assume that all agents can reach any location in the mission area. Thus, constraints on path feasibility were not considered in this study. In addition, regarding the characteristics of the agent itself, we considered two types of constraints in our research.

Each agent has an energy storage device. When the energy in the device is exhausted, the agent will not be able to move or participate in any mission group, and it will take some time to replenish the energy. We choose to use the maximum distance L_{max} that the agent could move to indicate the maximum capacity of the battery or fuel tank. Let $E_{imax} = L_{imax}$, E_{max} represents the maximum energy. Thus, the energy currently available on the agent E_{ic} can be expressed as:

$$E_{ic} = f_{ch} E_{imax} - L_{im}, \quad (1)$$

where L_{im} indicates the mileage of the agent i; f_{ch} represents the number of charges.

In the actual confrontation process, the amount of ammunition carried by the agent is limited, so in addition to energy constraints, ammunition constraints also should be considered. We translated the ammunition constraints into the number of missions in which the agent could participate. Let am_i indicates the remaining number of times that agent i can participate in the mission. When

$$\begin{cases} E_{ic} > 0 \\ am_i > 0 \end{cases} \quad (2)$$

the agent i is in a state that can be grouped.

2.2 Mission

The characteristics of the agent and the mission scenario have been introduced above. Next, some attributes of the mission are introduced.

In the combat environment of this paper, there are multiple missions at the same time, and each mission is independent. Due to the limited capabilities of the agents, when attacking enemy targets, they need to form groups to complete the mission. Because our study focused on the dynamic grouping mechanism of heterogeneous agents, we ignore the impact of the environment on its movement.

In order to improve the versatility of the model, the settings of the missions need to be as close as possible to the actual situation. During the simulation, the positions of some targets are unknown and need to be obtained through exploration. The continuous updating of the missions list ensures the dynamic nature of the grouping process. Besides, some targets are removable, which improves the authenticity of the model. Moreover, the mission should be completed within the specified time, when exceeding the time limit, it is considered to have failed.

- Mission characteristics

(1) Mission duration. The time elapsed from the generation of the mission to the announcement of the failure of the mission, denoted by t_d.

(2) Mission requirements for capabilities. Here, the capability requirement vector was used to represent a mission for each capability requirement. For a specific mission, the capability requirements matched the types of capabilities that all intelligent agents had, that is, the dimension of the vector is the same as the slot type of the agent. The vector of capability requirements could be expressed as:

$$D_k = [d_{k1}, d_{k2}, ..., d_{kp}, ..., d_{kn}], \tag{3}$$

where d_{kp} represents the demand for the pth capability of the mission k, and n is the type of capability.

(3) The time required for mission k to complete is expressed in t_{ck}. The sum of each ability requirement of the mission is positively correlated with the number of agents dispatched to perform this mission, so we fix the value of t_{ck}, which does not change with the needs of the mission.

Fig. 4 Basic mission flow

Fig. 4 shows the basic mission flow. When the missions are not completed, the mission group needs to be reconstructed according to the new mission requirements until all missions are completed.

- Constraints

We assume that there is a mission k and a corresponding mission group i, then the relationship between k and i meet the following condition:

$$\forall\, 1 \le j \le n, \quad \exists\, d_{kj} \le Cap(p_{ij}), \tag{4}$$

where j represents a certain ability, $Cap(p_{ij})$ represents the sum of the ability j in group i. This condition ensures that the mission can be executed smoothly.

2.3 The proposed model

(1) Objective 1: Mission rewards ($R(M)$)

In the process of forming a group, the sum of capabilities of the members in the group is required to be greater than the mission's demand for capabilities. According to the matching idea, certain principles should be satisfied for each mission group: the higher the degree of matching between the mission group's capabilities and the needs of the mission, the greater the benefit of mission completion (Shehory and Kraus, 1998). This is because, in the case of ensuring the completion of the mission, if a mission group whose ability far exceeds the demand is used to complete the mission, it will cause a waste of the agent's capability resources and reduce the overall profit of the mission. This section measures mission rewards $R(M)$ by the degree of ability matching. According to the above ideas, we give the numerical calculation method of $R(M)$ based on the matching degree:

$$R(M)=\sum_{k=1}^{n}P_k r_{Mk}, \tag{5}$$

where $R(M)$ represents the overall rewards of the mission, P_k represents the priority of mission k; r_{Mk} represents the reward of mission k based on the matching degree, and the calculation method of r_{Mk} is as follows:

$$r_{Mk}=\begin{cases} b_k-ne_k & accomplished \\ 0 & failed \end{cases}, \tag{6}$$

$$b_k=\gamma\sum_{j=1}^{n}d_{kj}, \tag{7}$$

$$ne_k=\frac{O_k D_{rk}^T}{\sum_{j=1}^{n}D_{rkj}}, \tag{8}$$

where b_k represents the ideal reward of mission k, that is, the benefits generated when the sum of the capabilities of the members of the group is exactly the same as the mission's needs for capabilities. b_k is measured by the sum of capability requirements of the mission D_k.

According to the relationship between the reward and the matching degree mentioned above, when the capabilities cannot be fully matched, the negative reward ne_k generated by the redundant part of the capabilities need to be subtracted from b_k. Equation (8) is used to calculate negative reward ne_k, where O_k represents the redundant part of the capability. It can be calculated by:

$$O_k=Cap(p_i)-d_k. \tag{9}$$

D_{rk} is a vector consisting of the reciprocal of each element in D_k, and $\frac{D_{rk}^T}{\sum_{j=1}^{n}D_{rkj}}$ is used as the weight coefficient of O_k to measure the impact of the overflow part of each capability on reward. When the mission is completed, the specific mission reward can be calculated; otherwise, $r_{Mk}=0$.

(2) Objective 2: Fuel cost ($F(M)$)

In addition to the rewards of the missions, we also have to calculate the cost of the missions. $F(M)$ represents the fuel cost generated during the movement of all agents and is described by the average moving distance of the agent. It can be calculated by:

$$F(M)=\frac{\sum_{i=1}^{p}L_{im}}{p}, \tag{10}$$

where L_{im} indicates the mileage of the agent i. p is the number of agents.

(3) Objective 3: Networking overhead ($E(M)$)

Besides, in the process of mission execution, periodic data interaction between individuals must be guaranteed by each mission group. What we can influence in the grouping algorithm is only the communication data and energy consumption generated during the networking process. In this section, we use energy consumption $E(M)$ as a parameter to measure the communication overhead and its impact when networking. The larger the value of $E(M)$, the greater the communication volume and energy consumed during networking, and the greater the cost of the mission.

In order to ensure stable data interaction during the mission, we choose a fixed distribution type TDMA (Time Division Multiple Access) as the method for nodes to access the network. We do not study the access protocol and data structure, only calculate the energy consumed by sending application data when the node uses the TDMA protocol to access the network. The following formula gives the calculation method:

$$E_{ipbit}=\sum_{j=1}^{N_{si}}\left[\left(P_{ct}+P_{cr}\right)/\zeta R_s+Td_{tj}^{n_t}\right]. \tag{11}$$

Formula (11) is given by (Cui et al., 2004) and (Jiang et al., 2010), and it is used to calculate the energy consumption of nodes transmitting data. Where E_{ipbit} represents the energy consumed by node i to the leader per 1-bit data transmission; N_{si} represents the hop number from node i to the group leader; P_{ct} and P_{cr} are transmitting circuit power and receiving circuit power respectively; ζ represents modulation parameters; R_s represents bit rate; Under the condition of point-to-point transmission, T can be regarded as a constant, which depends on the modulation form, circuit compensation, antenna power gain and other parameters; d_{tj} represents the transmission distance from node j to the next node in the transmission link.

Under the conditions of this paper, except for d_{tj} and N_{si}, the remaining parameters can be regarded as constants, and the values are given by (Jiang et al., 2010).

For all missions, the total energy consumption during the networking process is:

$$E(M)=\sum_{i=1}^{N_e}S_iE_{ipbit}, \tag{12}$$

where N_e is the number of times that all nodes are connected to the network, and S_i represents the total amount of application data sent by node i.

Based on the above description, the model is formulated as follows:

$$\begin{cases} \max & R(M) \\ \min & F(M) \\ \min & E(M) \end{cases}$$
$$\text{s.t. (2) and (4).}$$

We give three objectives in terms of mission rewards and mission costs. The decision variables include the sum of the capabilities of the mission group $Cap(p)$, the transmission distance d_t, the number of nodes applying to the network N_e, and the agent's mileage L_m. The value of the above decision variables depends only on the grouping scheme.

3 Dynamic group formation method based on utility function and heuristic rules

In the previous grouping method, the mission team was immediately disbanded after completing the mission, and then the decentralized agents formed a new group according to the mission requirements.

Unlike previous research, we introduced a "Dynamic adjustment" mechanism in the mission group restructuring strategy. Once the mission is completed, team members will be adjusted to meet the new mission needs by combining, absorbing new agents, group splitting, and other operations, instead of being disbanded immediately.

Individuals in the group share member and mission information, and each group moves and performs the mission as a whole.

3.1 Utility Function

Before grouping, we designed the utility function to measure the matching degree between the agent and the mission. The higher the value of the utility function, the more suitable the agent is to complete the mission. We use the calculation results as the basis for dynamic grouping.

When determining the utility function, we considered the following factors:

(1) Urgency of the mission k, defined as follows:

$$ur_k = \frac{1}{t_d - t_{ek}}. \tag{13}$$

(2) Euclidean distance. The distance d_{ik} from the agent to the mission k is also an important factor that affects whether the agent is suitable for performing the mission.

$$d_{ik} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}, \tag{14}$$

(x_i, y_i) and (x_k, y_k) are the coordinates of agent i and mission k.

(3) The evaluation value $conf_{ik}$ represents the evaluation given by the agent i on the mission k, which can be defined as follows:

$$conf_{ik} = e^{-(t-t_{fk})}, \tag{15}$$

where t represents the current moment, t_{fk} is the moment when mission j was discovered. The greater the value of $conf_{ik}$, the higher the probability that the agent believes that mission k can be completed.

Based on the above three factors, the utility function of agent i for mission k could be expressed as:

$$u_{ik} = \frac{\alpha \cdot ur_k + conf_{ik} \cdot \beta}{d_{ik}}, \tag{16}$$

where α and β are the weight coefficients of each parameter respectively.

Equation (16) can be used to calculate the utility value of each agent for mission i. For group j, the average utility value $\bar{u}_{jk}$ can be calculated to determine whether group j is suitable for performing mission k.

$$\bar{u}_{jk} = \frac{\sum_{i=1}^{n_j} u_{ik}}{n_j} \tag{17}$$

where n_j represents the number of agents in group j.

3.2 Heuristic rules for dynamic group formation

Next, a networking-overhead-based constructive heuristic self-organizing rule (NCH) is introduced. When designing the heuristic rules, we considered the network overhead and tried to maintain the original group staffing during the formation of the new mission group, in order to reduce the communication overhead when networking.

We use *mlist* to save the missions that need to be executed currently, that is, a list of missions. $|mlist|$ represents the number of missions in *mlist*.

Step 1: Sort current missions in *mlist* based on urgency.

Step 2: For $1 < k < |mlist|$

Based on the latest mission information recorded by the detection agent, determine the capability demand vector D_k of each mission.

Determine the capability requirements D_k of each mission in *mlist*, the form of D_k is given by formula (3).

Step 3: For $1 < k < |mlist|$

According to the existing grouping situation, the currently idle mission groups are counted to form *oglist*. The number of groups in *oglist* is represented by $|oglist|$.

Step 4: For $1 < j < |oglist|$

Select groups in order in *oglist*, and calculate the average utility value $\bar{u}_{jk}$ of each group for mission k according to Equation (17).

Step 5: Sort the groups in *oglist* according to the value of $\bar{u}_{jk}$ from high to low, and save the new group order in *glist*. The number of groups in *glist* is represented by $|glist|$, $|glist| = |oglist|$.

Step 6: Let $group_k$ be the group used to perform mission k. Based on the needs of mission k, we will select the appropriate members in *glist* to join $group_k$ to perform the mission k. The purpose of this step is to select multiple individuals which are most suitable for performing mission k to form a group while keeping the original mission group as much as possible.

For $1 < j < |glist|$

When selecting members to form the group of mission k, we will compare the capabilities of groups 1 to $|glist|$ with the demand D_k of k in the order of *glist*. According to whether (4) is satisfied, it is divided into the following two cases:

(a) If the relationship between the capabilities of group j and mission k does not satisfy (4), it means that group j does not meet the current needs D_k. Let all members of group j join $group_k$. The difference between the capabilities of j and k is calculated as the new D_k. And then return to step 6, $j = j + 1$.

(b) Conversely, if (4) is satisfied, it means that group j meets the current needs of mission k. At this time, if all the members of j join $group_k$, some of the individual's capability may be wasted. Therefore, we need to combine the improved response threshold method to select suitable individuals from j to join $group_k$ and avoid the waste of agents. The improved threshold model comes from (Kim et al., 2014), shown in Equation (18).

$$P(S_{uk}, \theta_{uk}) = \frac{S_{uk}^2}{S_{uk}^2 + a\theta_{uk}^2 + \mathit{\Delta}\tau_{uk}^{2b}}, \tag{18}$$

where S_{uk} represents the mission's stimulus for agent u, θ_{uk} is the threshold, τ_{uk} represents the time required for the agent u to reach the position of mission k, and a and b are parameters. The lower an agent's threshold or the higher a mission's stimulus, the more likely it was for the agent to accept the mission.

Mission k has different stimulus for different agents, it can be calculated by:

$$S_{uk} = \max\{D_k\} \cdot \mathrm{cap}\,(p_{uv}), \tag{19}$$

$cap\,(p_{uv})$ is the vth capability of agent u, and its type is the same as the type of capability most needed by mission k. If $\max\{D_k\} = d_{kn}$, then $v = n$ and $cap\,(p_{uv}) = cap\,(p_{un})$.

We let the agent choose mission k with probability $P(S_{uk}, \theta_{uk})$ every second. After each selection, let the individuals who choose k join $group_k$. There are also two cases at this time. When the relationship between $group_k$ and D_k satisfies the constraint (4), the grouping of mission k is completed. Otherwise, D_k and S_{uk} need to be updated. Repeat the above operation until $group_k$ and k meet the constraints (4). After obtaining $group_k$, the remaining agents in group j form a new group j and continue to participate in the grouping of subsequent missions.

Through the above operations, we incorporated the response threshold method into the heuristic framework, and effectively solved the problem of screening agents.

Step 7: Repeat steps $4-6$, until one of the following two conditions occurs, terminating the grouping process:

(a) All missions in the current mission list are performed by a certain group.

(b) When forming a group for mission k in the list, the remaining idle agents are not enough to perform that mission.

When situation (b) occurs, in order to save time, the remaining idle agents go to the vicinity of mission k and stand by.

Besides, all agents participating in the grouping must satisfy constraint (2).

During the grouping process, some of the original connections will be disconnected, and new connections will be formed at the new mission.

When choosing a leader for a new group, we try to choose the original leader included in the group, so that the connection between the leader and surrounding nodes can be maintained.

Through the above method, the dynamic grouping problem of agents can be solved. After the formation of the mission group, when the agent moves to the vicinity of the mission, if its distance from the leader or the nodes around the leader is less than the communication radius, it can send an application to join the network. We stipulate that the information transmission link from the member to the group leader should not exceed 2 hops at most.

4 Experiments

After the design and description of the model and the dynamic self-organizing method of the agents, we conducted a series of simulation experiments based on the self-organizing method designed in the study. We hoped to obtain the performance of the algorithm under different scales and different scenarios through experimentation.

The comparison algorithm selects the response threshold method introduced from (Kim et al., 2014), and the adjusted ACO based on the model in this paper.

- Adjusted ACO

Generally speaking, the ant colony algorithm is to set up the population in the mission environment and spread the pheromone along the way through the ants. In the problem of this paper, the pheromone needs to be set at the mission position to attract the agent to execute. The concentration of the pheromone of the kth mission is represented by τ_k, the probability that agent i chooses mission k can be calculated as follows:

$$P_{ik} = \frac{(\tau_k)^{\alpha d}}{\sum_{s=1}^{q}(\tau_s)^{\alpha d}}. \quad (20)$$

After a round of selection, if the needs of mission k are met, set τ_k to zero; otherwise update τ_k according to the following formula and continue to attract agents.

$$\tau_k = \tau_k + \Delta\tau, \quad (21)$$

$\Delta\tau$ is the concentration of increased pheromone.

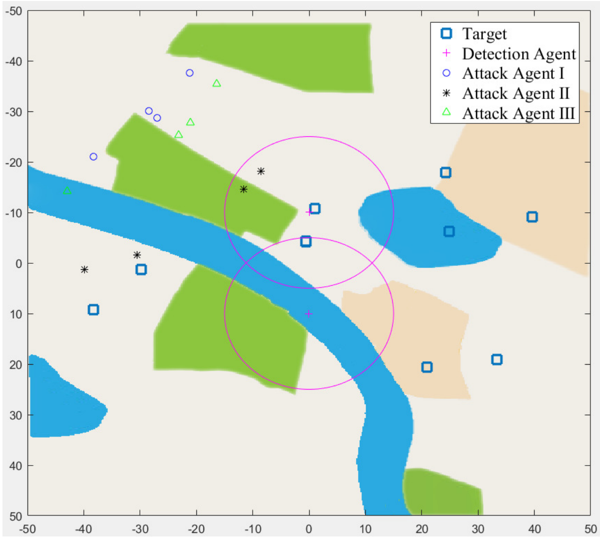


Fig. 5 Simulation scenario

4.1 Settings

Based on the problems studied in this paper, we designed four sets of comparative experiments to compare the application effect of the algorithm under the conditions of different mission numbers. Table 1 and 2 show the different values used for the test runs for all model parameters. The parameter values in Table 1 can be adjusted in the simulation, and the parameter values in Table 2 are derived from (Cui et al., 2004) and (Jiang et al., 2010). From these tables, we can see the specific parameter settings when we perform four sets of missions of different sizes in the same mission area.

In the simulation, we used one type of detection agent and three types of attack agent. The parameters of the four types of agent are given in Table 3. Some values in Table 3 refer to the relevant parameters of actual weapons and equipment. Table 4 sets the parameters of the algorithm for comparison.

- Simulation scenario

Fig. 5 shows the simulation scenario. The small squares represent the enemy deployment units, which were randomly generated in the mission area as the simulation advanced. The large circle indicates the detection agent and its detection range. After the simulation started, the detection agent looped

Table 1 Variable parameters in the experiment

Parameter	Definition	Scenario 1	Scenario 2	Scenario 3	Scenario 4
G	Mission area (km^2)	100*100	100*100	100*100	100*100
q	Number of enemy units	10	25	35	50
-	Enemy location	Random	Random	Random	Random
-	Type of agent	4	4	4	4
-	Type of Capability	3	3	3	3
δ_k	Importance of mission k	Random	Random	Random	Random
t_{sp}	Simulation step	1	1	1	1
t_{ck}	Time required to complete the mission	5	5	5	5
t_d	Mission duration	20	20	25	30
t_r	Replenishing time	2	2	2	2
α	Weight coefficient	10	10	10	10
β	Weight coefficient	5	5	5	5
γ	Weight coefficient	0.6	0.6	0.6	0.6

Table 2 Fixed parameters in the experiment

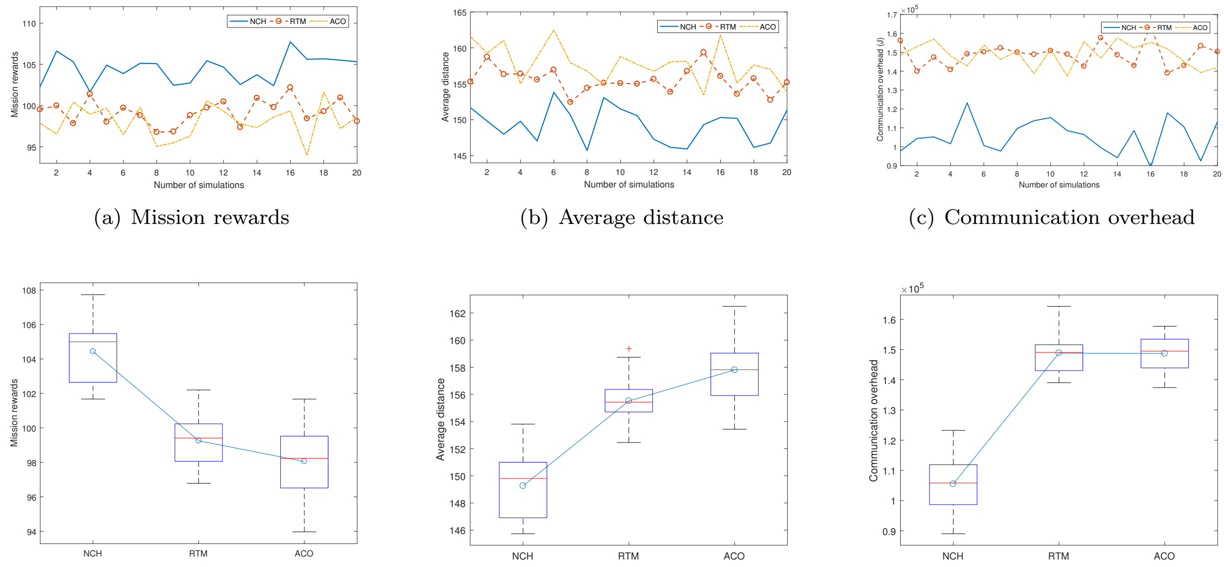
Parameter	Definition	Value
P_{ct}	Transmitting circuit power	98.2mW
P_{cr}	Receiving circuit power	112.5mW
ζ	Modulation parameters	1
T	Constant	10^{-18}
R_s	Bit rate	10^4 symbol/s
n_t	Constant	3
S_i	Weight coefficient	20 bit

Table 3 Parameters of agents

Agents	Velocity	Fuel	Ammunition	Anti-air	Ground Attack	Maneuverability	Quantity
Type1	300km/h	8000	-	-	-	10	2
Type2	100km/h	800	16	3	4	4	8
Type3	60 km/h	400	40	1	5	2	12
Type4	70 km/h	400	12	5	2	3	10

Table 4 Parameters of the algorithm to be compared

Algorithm	Parameter	Definition	Value
ACO	αd	Heuristic factor	1.5
	$\Delta\tau$	Pheromone increment	0.2
RTM(Kim et al., 2014)	Θ_{max}	Maximum threshold	40
	a	Weight coefficient	2
	b	Weight coefficient	1.5



(a) Mission rewards (b) Average distance (c) Communication overhead

(d) Mission rewards (e) Average distance (f) Communication overhead

Fig. 6 The number of missions is 10 (Scenario 1).

through the mission area to update the mission information. The remaining three symbols, which are triangles, asterisks, and small circles, represent three types of attack agents that performed strikes based on the grouping results from the edge of the mission area.

4.2 Results

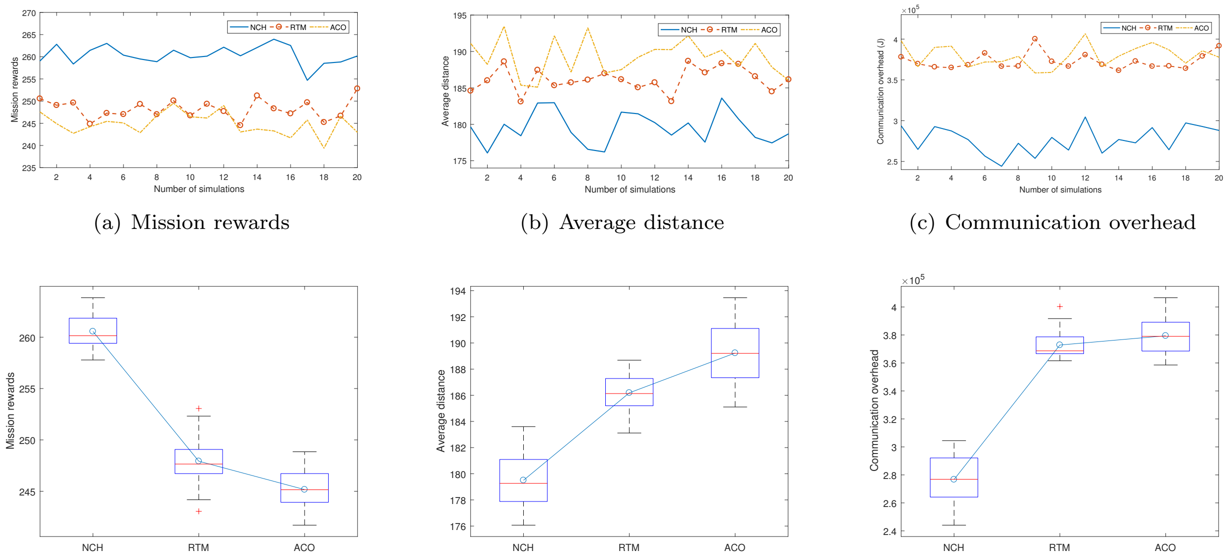
By simulating the four mission scenarios, we can compare the running results of the dynamic group formation strategy under different mission numbers.

In the following simulation results, the blue line represents the result of the dynamic grouping method (NCH) designed in this paper; the red line represents the result of the response threshold method (RTM); and the yellow line represents the result of the ant colony algorithm (ACO). We conducted 20 simulations on the four mission scenarios given in Table 1 respectively.

Figures 6 to 9 respectively show the simulation results obtained under four different mission scenarios, figure parts (a), (b), and (c) represent the results of three objectives under different algorithms. We performed simulations on missions of different sizes, and the results show that the NCH method is superior to the comparison algorithm in all three objectives; and as the amount of missions increases, the performance of NCH becomes more prominent. Fig. 8 and 9 clearly reflect the advantages of NCH compared to the other two comparison algorithms when the number of missions is large.

Moreover, figure parts (d), (e), and (f) are box plots of three objectives, which represent the mean value and fluctuation range of different objectives under different methods. Through the box plot, simulation and comparison results can be more intuitively reflected. As the box plots show, compared with the comparison algorithm, when NCH is used to dynamically form the mission group, the average of the three objectives is better. However, the advantages of NCH are not obvious in terms of the volatility of the solution results, which means that in terms of stability, our method (NCH) has room for improvement.

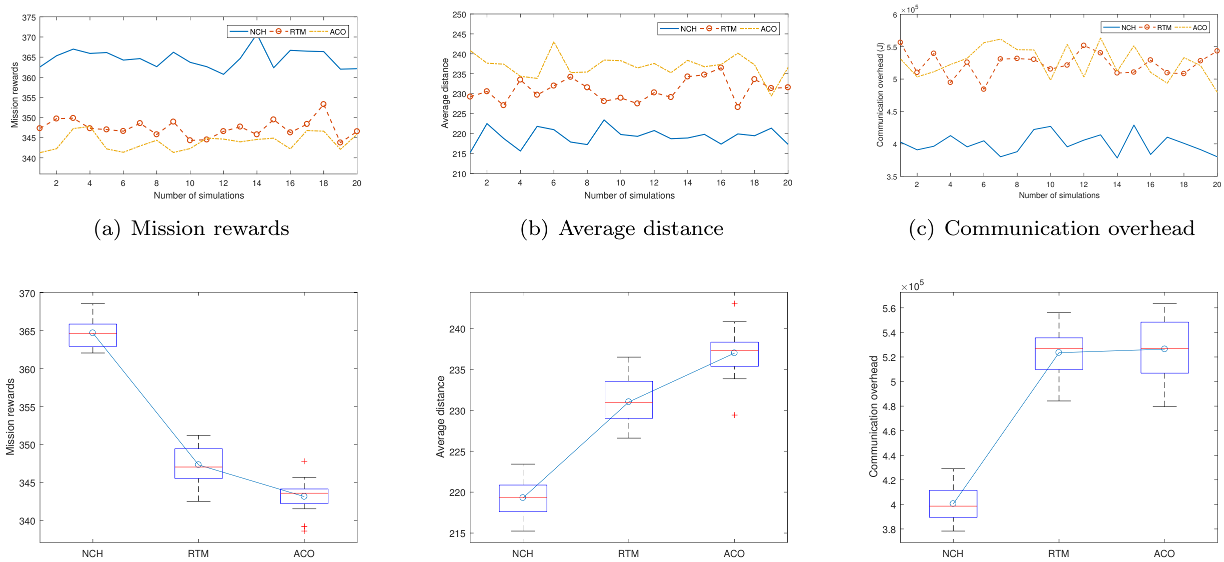
Table 5 is a specific grouping statistics of scenario 1. Through the grouping statistical results, it can be intuitively understood that since the heuristic rules of the NCH method consider the energy consumption factor, the original group member structure can be maintained as much as possible when the method is used for dynamic grouping. On the contrary, the membership of the mission group of the two comparison algorithms is more random. The comparison can prove the effectiveness of heuristic rules and NCH algorithm.



(a) Mission rewards (b) Average distance (c) Communication overhead

(d) Mission rewards (e) Average distance (f) Communication overhead

Fig. 7 The number of missions is 25 (Scenario 2).

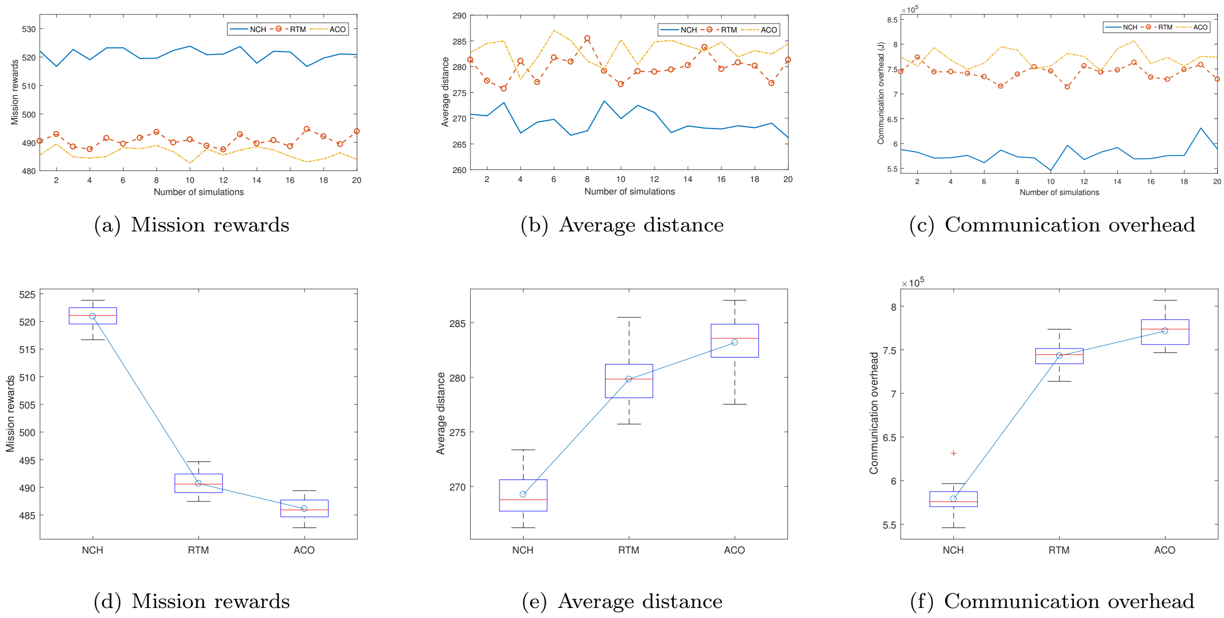


(a) Mission rewards (b) Average distance (c) Communication overhead

(d) Mission rewards (e) Average distance (f) Communication overhead

Fig. 8 The number of missions is 35 (Scenario 3).

In terms of the characteristics of the algorithm itself, the use of the NCH method is based on the layered distributed system designed in this paper. The implementation of heuristic rules also depends on some simple decisions made by the detection agent (gateway node), such as sorting groups according to to the utility value. Therefore, NCH is not completely a distributed algorithm, but combines some features of a centralized algorithm. The comparison algorithms ACO and RTM are distributed algorithms, which can completely realize the dynamic self-organization of agents without relying on superior nodes. In this respect, the performance of NCH is worse than them. In other words, NCH has certain advantages in solving the problems in this paper, but under other conditions, the performance of NCH



(a) Mission rewards (b) Average distance (c) Communication overhead

(d) Mission rewards (e) Average distance (f) Communication overhead

Fig. 9 The number of missions is 50 (Scenario 4).

Table 5 Statistics of grouping results (Scenario 1)

	T=10s		T=20s		T=30s	
	Mission	Agents	Mission	Agents	Mission	Agents
NCH	2	3,4,5,27,28,29,30	1	10,11,12,28	3	2,17,18,19,20,22,26
	4	2,17,18,19,20,26	7	1,13,14,15	10	3,4,5,23,24,25
	5	1,13,14,15,16	8	16,21		
	6	21,22,23,24,25	9	6,7,8,9		
ACO	2	2,4,5,18,26,28,29	1	7,10,12,13,21	7	1,6,7,8,13
	4	3,7,17,19,20,27	3	9,11,18,19,26,27,30	9	5,10,14,15
	5	1,11,14,23,25,30	8	17,22,29	10	3,4,9,20,24,28
	6	6,15,16,22,24,26				
RTM	2	1,4,13,24,27,28,30	1	11,12,16,23	7	1,12,13,17,18
	4	2,5,7,17,19,21,25	3	2,14,18,19,20,22,26	8	11,15,23
	5	3,14,15,16,18	9	6,7,8,9,17		
	6	6,20,22,23,26,29	10	3,4,5,10,13,24,25		

may not be as good.

5 Conclusion and future work

The purpose of our research is to design a heuristic mission group formation approach with some self-organizing characteristics according to the dynamic mission requirements. In the actual battlefield, frequent transmission of data may cause nodes to be detected, or consume too much energy and lose communication ability for a period of time. So We designed a series of heuristic rules to preserve the original group's organization as much as possible when forming a new group. This strategy effectively reduces the traffic generated by related steps by reducing the disconnection and reconstruction operations of links between nodes. In addition, based on the ability matching principle, we have also made adjustments to the existing self-organizing algorithm, reduced the waste of agent capabilities during the grouping process. The adjusted self-organizing algorithm and heuristic rules together form the mission group dynamic formation algorithm described in this paper.

In the simulation, we designed a mission scenario where heterogeneous agents search and attack enemy targets. Three objectives show that the NCH method has advantages in solving this problem.

In future work, the problem will be considered more complex. In actual combat, when different types of ammunition are carried, the capabilities of each agent will need to be reconfigurable. Also, the actual mission environment may contain many obstacles or unknown factors, which will affect the movement of agents and their group formation. Therefore, in the next step, we will study the dynamic grouping problem of agents based on the above new requirements and constraints.

In terms of the application of the method, the dynamic self-organizing method studied in this paper can be applied not only in the field of combat, but also to the grouping problem of other kinds of missions. In future work, we will transform the model and consider the characteristics of other agents to expand the application area of the proposed method.

References

A.H. MN, Kashan, 2019. An effective grouping evolution strategy algorithm enhanced with heuristic methods for assembly line balancing problem. *Journal of Advanced Manufacturing Systems*, 18(03):487-506.

Butenko S, Murphey R, Pardalos P, et al., 2013. Recent developments in cooperative control and optimization. Springer Science and Business Media.

Cui S, Goldsmith A, Bahai A, 2004. Energy-efficiency of mimo and cooperative mimo techniques in sensor networks. *IEEE Journal on selected areas in communications*, 22(6):1089-1098.

Ducatelle F, Caro GD, Gambardella L, 2010. Cooperative self-organization in a heterogeneous swarm robotic system. Proceedings of the 12th annual conference on Genetic and evolutionary computation, p.87-94.

George J, Sujit P, Sousa J, et al., 2010. Coalition formation with communication ranges and moving targets. Proceedings of the 2010 American Control Conference, p.28-32.

Gerkey B, Mataric M, 2004. A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, 23(9):939-954.

Guo M, Xin B, Chen J, et al., 2020. Multi-agent coalition formation by an efficient genetic algorithm with heuristic initialization and repair strategy. *Swarm and Evolutionary Computation*, 100686.

Hirsch M, Pardalos P, Murphey R, et al., 2008. Advances in Cooperative Control and Optimization. Springer Science and Business Media.

Hirsch M, Pardalos P, Commander C, et al., 2009. Optimization and Cooperative Control Strategies: Proceedings of the 8th International Conference on Cooperative Control and Optimization. Springer Science and Business Media.

Jiang R, Ji W, Zheng B, 2010. Joint optimization of energy consumption in cooperative wireless sensor networks. *Journal of Electronics and Information Technology*, 32(6):1475-1479.

Khan A, Aftab F, Zhang Z, 2019. Self-organization based clustering scheme for fanets using glowworm swarm optimization. *Physical Communication*, 36, 100769.

Khoshnoud F, Esat I, de Silva C, et al., 2019. Quantum network of cooperative unmanned autonomous systems. *Unmanned Systems*, 7(02):137-145.

Kim M, Baik H, Lee S, 2014. Response threshold model based uav search planning and task allocation. *Journal of Intelligent and Robotic Systems*, 75(3–4):625-640.

Liu M, Lin J, Yuan Y, 2013. Research of uav cooperative reconnaissance with self-organization path planning. International Conference on Computer, Networks and Communication Engineering, Atlantis Press:207-213.

Manathara J, Sujit P, Beard R, 2011. Multiple uav coalitions for a search and prosecute mission. *Journal of Intelligent and Robotic Systems*, 62(1):125-158.

Merabet G, Essaaidi M, Talei H, et al., 2014. Applications of multi-agent systems in smart grids: A survey. 2014 International conference on multimedia computing and systems (ICMCS), p.1088-1094.

Moritz R, Middendorf M, 2015. Decentralized and dynamic group formation of reconfigurable agents. *Memetic Computing*, 7(2):77-91.

Murphey R, Pardalos P, eds., 2002. Cooperative control and optimization. Springer Science and Business Media.

Necsulescu P, Schilling K, 2015. Automation of a multiple robot self-organizing multi-hop mobile ad-hoc network (manet) using signal strength. IEEE International Instrumentation and Measurement Technology Conference, p.505-510.

Oh G, Kim Y, J.Ahn, et al., 2018. Task allocation of multiple UAVs for cooperative parcel delivery. In: Advances in Aerospace Guidance, Navigation and Control. Springer, Cham.

Orfanus D, de Freitas E, Eliassen F, 2016. Self-organization as a supporting paradigm for military uav relay networks. *IEEE Communications Letters*, 20(4):804-807.

Padmanabhan M, Suresh G, 2015. Coalition formation and task allocation of multiple autonomous robots. 2015 3rd International Conference on Signal Processing, Communication and Networking, p.1-5.

Pardalos P, Grundel D, Murphey R, et al., 2008. Cooperative networks: control and optimization. Edward Elgar Publishing, Incorporated.

Pardalos P, Butenko S, Murphey R, et al., 2013. Cooperative Control: Models, Applications and Algorithms. Springer Science and Business Media.

Ramchurn S, Polukarov M, Farinelli A, et al., 2010. Coalition formation with spatial and temporal constraints. Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems, p.1181-1188.

Shehory O, Kraus S, 1998. Methods for task allocation via agent coalition formation. *Artificial intelligence*, 101(1–2):165-200.

Singh V, Husaini S, Singh A, 2010. Self-organizing agent coalitions in distributed multi-agent systems. International Conference on Computational Intelligence and Communication Networks, p.650-655.

Skorobogatov G, Barrado C, Salamí E, 2020. Multiple uav systems: a survey. *Unmanned Systems*, 8(2):149-169.

Vig L, Adams J, 2006. Multi-robot coalition formation. *IEEE Transactions on Robotics*, 22(4):637-649.

Yang Y, Qiu X, Meng L, et al., 2014. Task coalition formation and self-adjustment in the wireless sensor networks. *International Journal of Communication Systems*, 27(10):2241-2254.